REMARKS

In response to the above-identified Office Action, Applicants amend the Application and seek re-consideration in view of the following remarks. In this Response, Applicants amend claims 1-18 and 21-22. Applicants do not cancel or add any claims. Accordingly, claims 1-18 and 21-22 remain pending in the Application.

I.   Claims Rejected Under 35 U.S.C. §102

Claims 1, 7, and 12 stand rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,983,446 issued to Charisius et al. ("*Charisius*"). Applicants respectfully traverse the rejection, at least in view of the amendments to claims 1, 7, and 12.

To anticipate a claim, the cited reference must disclose each and every element of the rejected claim (*see* MPEP § 2131). Among other elements, independent claim 1 defines a method for verifying a generated computer code having a first plurality of lines generated from a model file, comprising:

> processing the model file to **generate** an expected computer code having a second plurality of lines from the model file, each line in the second plurality of lines corresponding to a line in the first plurality of lines; and
>
> comparing each line in the first plurality of lines and each corresponding line in the second plurality of lines to determine if the generated computer code and the expected computer code match. (Emphasis added).

Applicants submit that *Charisius* fails to disclose at least these elements of claim 1.

In making the rejection, the Patent Office characterizes *Charisius* as disclosing each of the limitations of claim 1; however, Applicants submit that *Charisius* fails to disclose two different computer codes (i.e., generated computer code and expected computer code) being generated from the same model file and each line of the generated computer code being compared to a corresponding line in the expected computer code to verify that the generated computer code is

correct.

> *Charisius* discloses a software development tool that provides:

> > simultaneous round-trip engineering, i.e., the graphical representation 204 is synchronized with the textual representation 206. Thus, if a change is made to the source code 202 via the graphical representation 204, the textual representation 206 is updated automatically. Similarly, if a change is made to the source code 202 via the textual representation 206, the graphical representation 204 is updated to remain synchronized. There is no repository, no batch code generation, and no risk of losing code. (Col. 5, lines 50-60).

Therefore, Applicants submit that *Charisius* discloses a system and method for automatically updating a graphical representation of source code when changes are being made to the source code itself and updating the source code when changes are being made to the graphical representation of the source code. In other words, *Charisius* discloses a system where the source code is automatically updated as the model is changed and the model is updated as the source code is changed. Thus, Applicants submit that there is only one computer code being generated in *Charisius'* system and method.

By contrast, claim 1 recites a method where each line of code in the <u>generated computer code</u> is compared to a corresponding line of code in expected computer code that is <u>generated</u> to verify that the generated computer code is correct. That is, a second computer code (i.e., expected computer code) is being generated and compared to a first computer code (i.e., generated computer code) to verify that the first computer code is correct, which is completely different from a system and method for keeping a graphics model and a source code in synch with one another.

Furthermore, the Patent Office cites the UML model class definitions and object-oriented constructs as reading on Applicants' ***expected code*** (*see* <u>Paper No./Mail Date 20080114</u>, page 2). Applicants disagree with this characterization of the UML model class definitions and object-oriented constructs.

Applicants submit that one skilled in the art recognizes that UML is a general-purpose modeling language where the UML model class definitions are predefined definitions, templates,

11

and syntax that do not change from application to application. Similarly, object-oriented constructs are predefined, set syntax for a particular programming language. That is, the object -oriented constructs do not change from application to application for a given programming language.

By contrast, the expected computer code recited in claim 1 will change as the contents of the model file change. That is, processing different model files results in different expected computer code. Moreover, claim 1 recites that the model file is processed to generate the expected computer code, whereas UML model class definitions and object-oriented constructs are not generated, but rather are already known and are simply loaded on the computing unit.

Furthermore, UML model class definitions and object-oriented constructs are not compared to corresponding UML model class definitions and object-oriented constructs in another program file to determine if the original UML model class definitions and object-oriented constructs are correct. Specifically, a compiler, for example, checks lines of code using UML model class definitions and object-oriented constructs to determine if the syntax of each line of code is correct in accordance with the rules of the particular programming language being used, but does not compare, for example, operators, functions, inputs, outputs, etc. in each line of code in a generated computer code and lines of code in an expected computer code generated from the same model file because the compiler does not "care" what the content of the operators, functions, inputs, outputs, etc. are, but rather, is only concerned with whether the operators, functions, inputs, outputs, etc. are written with the correct syntax. That is, the compiler is not "expecting" anything because the compiler only checks to determine if the UML model class definitions and object-oriented constructs syntax is proper for the particular programming language. By contrast, in claim 1 each line in the first plurality of lines of code are compared to each corresponding line in the second plurality of lines of code to determine if the lines of code match (i.e., determine that the respective lines of code are the same). Thus, the second plurality of lines of code is "expected" to be the same as the first plurality of lines of code, or else an error message is displayed.

The failure of *Charisius* to disclose each and every element of claim 1 is fatal to the anticipation rejection. Therefore, claim 1 is not anticipated by *Charisius*. Accordingly, Applicants respectfully request withdrawal of the rejection of independent claim 1.

Applicants submit that independent claims 7 and 12 each recite elements similar to the elements of "processing the model file to generate an expected computer code having a second plurality of lines from the model file, each line in the second plurality of lines corresponding to a line in the first plurality of lines, and comparing each line in the first plurality of lines and each corresponding line in the second plurality of lines to determine if the generated computer code and the expected computer code match," as recited in claim 1. Therefore, Applicants submit that claims 7 and 12 are not anticipated by *Charisius* at least for the same reasons as claim 1, in addition to their own respective features. Accordingly, Applicants respectfully request withdrawal of the rejection of claims 7 and 12.

II.   Claims Rejected Under 35 U.S.C. §103

Claims 2-6, 8-11, 13-18, and 21-22 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Charisius* in view of the ordinary skill in the art. Applicants traverse the rejection.

To render a claim obvious, the cited reference must teach or suggest each and every element of the rejected claim (*see* MPEP § 2143). Claims 2-6 and 21 depend from claim 1, claims 8-11 and 22 depend from claim 22, and claims 13-18 depend from claim 12 include all of the elements of their respective independent claims. Applicants have discussed above the failure of *Charisius* to teach or suggest each and every element of independent claims 1, 7, and 12, and submit that such discussion is equally applicable to claims 2-6, 8-11, 13-18, and 21-22 because of their respective dependencies from claims 1, 7, and 12. The Patent Office relies on the ordinary skill in the art to cure the defects of *Charisius*; however, Applicants submit that the ordinary skill in the art fails to cure such defects.

In making the rejection, the Patent Office does not rely on the ordinary skill in the art as teaching or suggesting the elements of, "processing the model file to generate an expected computer code having a second plurality of lines from the model file, each line in the second plurality of lines

13

corresponding to a line in the first plurality of lines, and comparing each line in the first plurality of lines and each corresponding line in the second plurality of lines to determine if the generated computer code and the expected computer code match," as recited in each of claims 2-6 and 21-22 via claim 1, and similarly recited in claims 8-11 and 13-18 via claims 7 and 12, respectively. Moreover, Applicants submit that such elements were not obvious to one of ordinary skill in the art prior to Applicants' earliest effective date. Therefore, the ordinary skill in the art fails to cure the defects of *Charisius*. Accordingly, Applicants respectfully request withdrawal of the rejection of claims 2-6, 8-11, 13-18, and 21-22.

CONCLUSION

In view of the foregoing, it is believed that all claims now pending are in condition for allowance. A Notice of Allowance is earnestly solicited at the earliest possible date. If the Examiner believes that a telephone conference would be useful in moving the application forward to allowance, the Examiner is encouraged to contact the undersigned at (480) 385-5060 or jgraff@ifllaw.com.

If necessary, the Commissioner is hereby authorized to charge payment or credit any overpayment to Deposit Account No. 50-2091 for any fees required under 37 C.F.R. §§ 1.16 or 1.17, particularly extension of time fees.

Respectfully submitted,

Date <u>March 17, 2008</u>   /JASON R. GRAFF REG. NO. 54,134/___
                                            Jason R. Graff
                                            Reg. No. 54,134